Remarks

This Request for Reconsideration is responsive to the final Office Action mailed October 20, 2005. In that Office Action, claims 1, 8 and 9 were finally rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,539,381 (hereinafter, "Prasad"); claims 2, 3, 6 and 7 were finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Prasad in view of U.S. Patent No. 5,944,840 (hereinafter, "Lever"); and claims 4 and 5 were finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Prasad and Lever combined with U.S. Patent No. 6,449,739 (hereinafter, "Landan"). Applicant respectfully requests reconsideration of these rejections under 37 C.F.R. §1.116 and passage of this application to allowance in view of the following remarks.

A. Anticipatory Rejections: Prasad

Keeping in mind that a claim can only be rejected as anticipated by a prior art reference if each and every limitation "as set forth in the claim" is found in the reference, MPEP §2131 (citing *Verdegaal Bros. V Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d. 1051, 1053 (Fed.Cir. 1987), Prasad simply does not provide support for a proper anticipatory rejection of any claim in this application. Prior to addressing these distinctions in detail, the present invention and the teachings of Prasad are briefly contrasted with one another.

In general, the present invention is directed to monitoring data updates in a replicated server environment in which data maintained on one server is replicated to various other servers. In order to track whether updates to the replicated data are carried over to its replicas on the other servers, the present invention involves maintaining a "replica partner vector" table on each server in the environment. Each of the replica partner vector tables includes an update sequence number for each other server in the environment or a timestamp indicating the last successful replication attempt with each other server. The update sequence numbers, each of which uniquely identifies a data update on a particular server in sequential fashion, provide a mechanism for indicating whether the update(s) to the data have been replicated to the other servers. Generally speaking, the servers exchange their replica partner vector tables with one another in order to determine whether changes made to data on one server have been applied to all replications of that data on the other servers.

Like the present invention, Prasad is directed to the same general theme of server replication. With specific regard to the passages cited in the Final Office Action, Prasad

describes methodology by which each server that stores a replica of a replica set in a distributed environment is provided functionality for tracking the state of each replica in the set. Prasad, at Col. 1, lines 5-7. To accomplish this, a plurality of timestamps associated with each server that maintains a replica in the replica set is stored on each server. Prasad, at Col. 1, lines 44-46. In response to a change to a local replica maintained on one of the servers, Prasad's process involves comparing a timestamp of the local replica to a timestamp associated with another one of the servers to determine whether the change has been applied to the replica maintained on that server. Prasad, at Col. 1, lines 42-51. If not, Prasad teaches updating the replica on that server and associating a new timestamp with that replica to indicate the recent update. Prasad, at Col. 1, lines 52-56.

Turning now to the claims, the present application includes two independent claims – claim 1 and 9, both of which are directed to monitoring replication latency in a distributed server environment. While claim 1 is directed to a method, claim 9 is directed to a computer-readable medium on which computer-executed components for practicing the method are stored thereon. The method embodied in claims 1 and 9 involves a first server (referred to as a "local" server) requesting replication from a second server (referred to as a "remote" server). To accomplish this, both claims recite the local server transmitting a copy of its replica partner vector table to the remote server. Upon receipt of the replica partner vector table from the local server, both claims further recite the remote server comparing the update sequence numbers and the timestamps identified therein to the update sequence numbers and timestamps in the replica partner vector table stored on the remote server. The method recited in claims 1 and 9 then involves the remote computer updating the received table and transmitting the updated table and object updates to the local server such that the local server may improve the update sequence numbers and time stamps in its replica partner vector table.

In contrast, Prasad teaches a process by which replication is effectuated by each individual server initiating the replication process in response to an update to data stored *thereon*. Indeed, Prasad does not teach a first server requesting another server to evaluate whether the first server requires replication to become current with all other replicas in the replica set. Rather, the replication process taught by Prasad is proactive by each server in response to a change to a replica stored on that same server. See, e.g., Prasad, at Col. 3, lines 44-51 ("...responsive to a change in a local replica at a first server,"). Therefore, Prasad necessarily cannot teach the

initial transmitting act of claims 1 and 9, i.e., "transmitting a copy of the replica partner vector table of a local server to a remote server."

Moreover, both claims 1 and 9 explicitly recite the maintenance of an update sequence number on each server that uniquely identifies each update or change to data, thereby providing a mechanism for improved evaluation as to whether replicas of the data on other servers are up to date. In contrast, Prasad is completely silent as to the use of an update sequence number or any other indicia for uniquely identifying updates to data on a server *according to a sequential reference scheme*. Prasad instead teaches determining whether a replica should be updated based on a comparison between time stamps and nothing more. With that said, Prasad's failure to teach such evaluations using a sequential approach in combination with the strict reliance on a time-based process necessarily suggests against the use of update sequence numbers.

Claims 1 and 9 additionally recite an act by which the remote server compares the update sequence numbers and time stamps in the received replica partner vector table to the update sequence numbers and time stamps in its own replica partner vector table. By reciting both such analyses (i.e., sequential and time-based comparison), claims 1 and 9 necessarily distinguish the claimed invention from the teachings of Prasad. Certainly Prasad cannot be considered to teach or even provide suggestion for such a dual-scheme comparison act by its deliberate focus solely on time-based evaluation.

Even further, claims 1 and 9 recite an act in which the local server calculates a difference between the timestamp for each server in the replica partner table and *a current time*. Contrasting Prasad, the only calculations made with respect to the disclosed time stamps involve comparing a first time stamp to a second time stamp. Prasad fails altogether to teach comparing a time stamp to a *current* time and, consequently, fails to teach the recited calculating act. Prasad also fails to provide any motivation for performing such an act by failing altogether to suggest the need for checking latency relative to an update to data on a server and replication of that update to any replicating servers. At most, Prasad briefly notes that replica synchronization may be performed periodically, see Prasad, at Col. 14, lines 9-11, but otherwise is silent as to any process by which a determination is made as to whether periodic synchronization has failed. Prasad consequently does not provide any suggestion for modification or combination of its teachings in the direction of the present invention relative to the recited calculating act.

For at least these reasons, claims 1 and 9 are believed allowable over Prasad, as are claims 2-8, each of which depend directly or indirectly from claim 1.

B. Obviousness Rejections: Lever

Unlike Prasad, Lever is not directed to server replication. Instead, Lever is directed to a completely different technology that relates to measuring interrupt latency. See Lever, at Col. 1, lines 5-9. In this regard, Lever determines the amount of time (i.e., "temporal time") that it takes a computer to process an interrupt and compares this temporal time to a current time to render a "latency time." See Lever, at Col. 2, line 62 - Col. 3, line 3. Lever then compares the latency time to a predetermined threshold to determine whether an alarm or alert should be provided to a user of the computer. See Lever, at Col. 3, lines 39-41. It is with respect to such a comparison that Lever is being used in combination with Prasad to render claims 2-7 obvious.

The combination of Lever and Prasad does not amount to a prima facie case of obviousness over claim 2 or any claim for that matter. First and foremost, this combination fails to teach or suggest each and every limitation recited in any claim of the present application. See MPEP §2143.03 (citing *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (CCPA 1974). As noted above, Prasad is deficient at least with respect to the initial transmitting act, the comparing act and the calculating act recited in each of claims 1 and 9 in addition to failing to teach update sequence numbers. Lever does not teach these acts and features missing from Prasad and, moreover, is directed to a completely separate technology that would have no purpose even practicing such acts or utilizing an update sequence number. Claims 2-7 further teach an act of comparing the difference calculated in claim 1 to a maximum allowable latency time period. This "difference" is explicitly defined to be a difference between the timestamp for each server in the replica partner table and a current time. Lever is in no way directed to server replication and consequently cannot teach a server replication partner table having timestamps, from which any differences are determined. Therefore, the combination of Prasad and Lever additionally fails to teach the comparing act recited in claim 2.

Furthermore, the Office Action fails to establish a suggestion or motivation in either Lever or Prasad, or in the knowledge generally available to one of ordinary skill in the prior art, to combine these two references. See MPEP §2143.03 (citing *In Re Rouffet*, 149 F.3d 1350, 1357, 47 U.S.P.Q.2d 1453, 1457-58 (Fed.Cir. 1998)). Plainly stated, Prasad and Lever are

directed to two completely separate technologies - server replication and computer interrupt processing. These technologies are not even analogous with respect to one another and the Office Action fails to provide any evidence whatsoever that one of skill in the art would have recognized the motivation to combine these references. In addition to the fact that Lever does not teach the comparing act of claim 2, see supra, there simply is no suggestion or motivation in this reference to modify the latency evaluation relative to interrupts in the direction of server replication. The only plausible explanation for such modification is the impermissible use of hindsight reconstruction that appears to be based on the problems solved by the Applicant's invention. See, e.g., Ex parte Haymond, 41 U.S.P.Q.2d 1217, 1220 (BdPatApp&Int 1996) (the examiner "may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.).

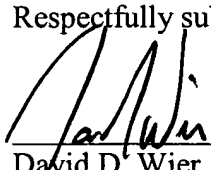
To summarize, short of improper hindsight, there simply is neither a logical explanation for modifying Lever's latency evaluation to apply to server replication nor for combining Lever with Prasad. For at least theses reasons as well, the combination of Lever and Prasad cannot support a prima facie case of obviousness with respect to claim 2 or any other claim in the present application.

Conclusion

This Request for Reconsideration is believed to be fully responsive to the Final Office Action mailed on October 20, 2005. Still, the Final Office Action contains arguments that are not directly addressed by this Request due to the fact that they are rendered moot in light of the preceding arguments in favor of patentability. Failure of this Request to directly address an argument raised in the Final Office Action should not be taken as an indication that the Applicant believes the argument to have merit. Furthermore, the claims of the present application may include other elements, not discussed in this Request, which are not shown, taught, or otherwise suggested by the art of record. Accordingly, the preceding arguments in favor of patentability are advanced without prejudice to other bases of patentability.

As all claims are believed clearly allowable over the art of record for the reasons stated herein, prompt reconsideration and allowance of this application are earnestly solicited. Should the Examiner have any remaining questions or concerns, he is encouraged to contact the undersigned attorney by telephone to expeditiously resolve such concerns. No fees are believed due for the submission of this Request. However, if this is not the case, please charge any such fees, including any extension fees due under 37 C.F.R. §1.136(a), to Deposit Account No. 13-2725. Alternatively, please credit any overpayment to Deposit Account No. 13-2725.

Respectfully submitted,



Dated: December 20, 2005

David D. Wier, Reg. No. 48,229
MERCHANT & GOULD P.C.
P. O. Box 2903
Minneapolis, MN 55402-0903
(303) 357-1647